ESD-TR-76-186

Semiannual Technical Summary

Information Processing Techniques Program

Volume I: Packet Speech/Acoustic Convolvers

30 June 1976

Prepared for the Defense Advanced Research Projects Agency under Electronic Systems Division Contract F19628-76-C-0002 by

Lincoln Laboratory

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LEXINGTON, MASSACHUSETTS

Approved for public release; distribution unlimited.

The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology. This work was sponsored by the Defense Advanced Research Projects Agency under Air Force Contract F19628-76-C-0002 (ARPA Orders 2006 and 2929).

This report may be reproduced to satisfy needs of U.S. Government agencies.

The views and conclusions contained in this document are those of the contractor and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency of the United States Government.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Raymond L. Loiselle

Raymond L. Loiselle, Lt. Col., USAF
Chief, ESD Lincoln Laboratory Project Office

Non-Lincoln Recipients

PLEASE DO NOT RETURN

Permission is given to destroy this document when it is no longer needed.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

INFORMATION PROCESSING TECHNIQUES PROGRAM
VOLUME I: PACKET SPEECH/ACOUSTIC CONVOLVERS

SEMIANNUAL TECHNICAL SUMMARY REPORT
TO THE
DEFENSE ADVANCED RESEARCH PROJECTS AGENCY

1 JANUARY – 30 JUNE 1976

ISSUED 23 AUGUST 1976

LEXINGTON MASSACHUSETTS

ABSTRACT

This report describes work performed under three programs: Packet Speech, Acoustic Convolvers, and Airborne Command and Control sponsored by the Information Processing Techniques Office of the Defense Advanced Research Projects Agency during the semiannual period 1 January through 30 June 1976. The first two programs are reported in Vol. I and the third in Vol. II.

CONTENTS

INFORMATION PROCESSING TECHNIQUES PROGRAM

I. PACKET SPEECH

A. SUMMARY

Packet speech conferencing on the ARPANET has been demonstrated with LPC speech, and the necessary software for continuously variable slope delta (CVSD) conferencing is now operational.

A preliminary investigation of the issues involved in permitting conversational speech and conferencing in a packet satellite network has been carried out. This study has led to an initial specification for a satellite conference simulation facility. Work on this simulation facility is under way.

In the user interface area, an initial version of a recognizer for spoken phrases has been implemented and undergone limited testing. A technique for deriving the necessary tables for syllable scoring directly from simple and easily updatable training statistics has been introduced. A phrase scorer has been developed which allows recognition of a key phrase within a more or less arbitrary carrier sentence. An interactive training facility has been implemented. A system structure has been specified for an initial implementation of a voice-controlled ARPANET mail retrieval system.

A new frequency-domain pitch extraction algorithm has been developed. Preliminary results indicate a substantial improvement over time-domain techniques when the input is telephone-quality speech.

Initial specifications for a charge-coupled device (CCD) homomorphic vocoder have been developed, and studies of channel vocoder filter bank implementation via CCDs have begun.

B. ARPANET CONFERENCING

Network conferencing has been demonstrated using linear predictive coding (LPC) vocoded speech and with participants at Information Sciences Institute (ISI), Culler-Harrison Incorporated (CHI), Stanford Research Institute (SRI), and Lincoln. The first conference experiments used a CHAIRMAN program running at Lincoln. In subsequent experiments, CHAIRMAN programs at CHI and ISI have been used successfully.

Recent effort in ARPANET conferencing has concentrated on the integration of software in the LDVT (Lincoln Digital Voice Terminal) and the PDP-11 to allow CVSD speech coding to be used in network conferencing. The software is now operational and has been used successfully in local conferencing (all participants at Lincoln). Echoing experiments in which the data are sent to ISI and returned to Lincoln indicate that the ARPANET can handle the 16-kbps data streams required for reasonable quality CVSD speech encoding.

C. PACKET SATELLITE SPEECH

A preliminary investigation has been undertaken to assess the outlook for conversational speech and conferencing in a packet satellite network. Input to the study consisted of various documents pertaining to packet satellites in general and some describing plans for the ARPA Packet Satellite Network in particular. This section reviews the results of the investigation and suggests some simulations which we propose to undertake. The approaches we suggest for handling speech are based on our interpretation of the current plans for the network. As the network develops, more information on its actual capabilities should become available, and we expect that we will need tc refine and extend the simulations to take account of such further knowledge.

The ARPA Packet Satellite Network will consist of two or three Satellite IMPs (SIMPs) sharing a single 56-kbps satellite channel. A number of different schemes for allocating the channel capacity among the SIMPs are being considered, but all of them seem to assume a slotted, time division handling of the channel which will give fixed-length packets carrying about 1200 bits. The propagation time up to the satellite and back is about 0.27 sec and 12 packet times will elapse between the transmission and reception of any individual packet. Since it is not yet clear exactly how many bits will be available to users, and since some of the allocation schemes make use of a round-trip frame concept, it seems reasonable to assume as a rule-of-thumb approximation that 16-kbps CVSD speech would require 4 of the possible 12 packet slots available in a frame to carry a speech data stream. LPC speech would need only one slot.

Since we are interested in looking at the particular problems posed by the satellite net, we have assumed that our speech processor can be connected directly to a host on a SIMP and, therefore, have direct access to the satellite channel. We have further assumed that the SIMPs involved in a conversation or conference will provide appropriate services such as the delivery of broadcast messages to multiple hosts. Available documentation does not indicate the extent to which such services are currently contemplated. We view it as one of the goals of the study to determine what special services would be desirable for speech users of a packet satellite net.

Delays in the satellite network result both from the relatively long inherent propagation time (0.27 sec) and from the algorithm used to allocate the channel. If the allocation scheme could operate perfectly (i.e., slots were always available when there were speech packets to transmit), a one-way delay of about 0.35 sec for CVSD speech (0.56 for LPC) could be achieved. These times result from adding the propagation time to the time required to accumulate a packetsworth of speech data. No additional smoothing delay would be needed since the jitter in arrival times would be negligible. These times are better than the ARPANET for CVSD and just about equal for LPC. A good realizable allocation algorithm (one which reserved an adequate number of slots at fixed times in the 12-slot frame) would introduce a variation in delay which would depend upon the difference between the time of generation of a speech packet and the time at which its reserved slot became available. This time difference would add a maximum of 1/4 of a round-trip time for CVSD and a full round-trip time for LPC. The delay would be fixed for the duration of a speech stream and, again, further smoothing delay would not be required. If the reserved slots could move around in the 12-slot frame or were not evenly spaced in the CVSD case, some additional smoothing delay would be required. We might thus expect that conversational (round-trip) delays could be in the range of 0.7 to 0.85 sec for 16-kbps CVSD speech and between 1.12 and 1.66 sec for LPC.

The above delay figures assume that an allocation scheme is available which would permit a stream reservation to be made and maintained for the duration of the conversation or conference. Clearly, the use of such reservations would be very inefficient if each speaker had to have a stream reservation in effect simultaneously so that his speech could be handled satisfactorily if he happened to produce any. There appear to be two approaches to be investigated which could minimize the wastage of channel capacity. One would be to relinquish the reservation at the end of each talkspurt, and establish a new reservation at the start of the next talkspurt, which might be produced by a different speaker at another site. The second approach would be to retain the reservation for the purposes of the conversation or conference but allow it to be handed off to another site in the event that another speaker wished to talk.

One of the allocation schemes which fits the first approach is called Reservation-ALOHA.* In this scheme, a SIMP is allowed to transmit a packet or not on the basis of the status of the most recently received packet slot. If that slot contained a packet successfully transmitted by the SIMP in question, or if the slot was empty (i.e., it did not contain a successfully received packet), then the SIMP would be permitted to transmit a new packet. In the event that the received packet "belonged" to the SIMP, the new packet would be transmitted without contention by other SIMPs since they would not (except for some small probability of error) allow themselves to transmit. If the slot had been empty, more than one SIMP might decide to transmit, and in that event a collision would occur, and the slot would be perceived as empty when it returned from the satellite. If the SIMP continued to transmit as opportunities arose, it could build up a set of "reserved" slots within the round-trip frame. The time to build up an adequate reservation (4 slots per frame for CVSD) would depend upon the activity of other users. In the case of data transmission, the SIMP would retransmit packets for which collisions occurred, but for speech, the added delay which would result from waiting for the retransmissions may be more disturbing than the glitches which would result from simply ignoring the collided packets. Once established at an adequate level, the reservation would permit glitch-free speech throughout the remainder of the talkspurt. When the talkspurt ended, the reservation would die away since there would be no new packets to transmit in the reserved slots.

The second allocation scheme which seems appropriate to the first approach of establishing and relinquishing reservations is the reservation scheme proposed by Roberts.† In this scheme, certain slots are set aside a priori for the transmission of reservation requests by an ALOHA technique. Such slots are subdivided since the reservation request message is much shorter than the data packet length. When a reservation request is successfully received, all SIMPs use the same algorithm to set aside future data packet slots to meet the requirements of the request. Plans for the packet satellite net indicate that some version of the Roberts scheme will be explored. It is to be expected that delay would be encountered in setting up a reservation depending upon other activity in the net. Unlike the situation with Reservation-ALOHA, the Roberts scheme allows no data transmission at all until the reservation has been established. At that point, however, speech would be essentially glitch free.

The second approach to handling speech without wasting capacity involves handing off the reservation from one speaker to another. This approach could work with any allocation scheme which allowed stream reservation. Such a reservation would be established in the process of setting up the call. Any delay in acquiring the reservation would be absorbed in the setting up process and would not affect the speech which came later. This approach seems well suited to the single data stream conferencing situation we have been exploring with experiments on the ARPANET. However, it would force simple conversational situations into a half-duplex mode,

*W. R. Crowther, R. Retteberg, D. Walden, S. Orstein, and F. Heart, "A System for Broadcast Communication Reservation-ALOHA," Proc. Sixth Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 1973.

†L. G. Roberts, "Dynamic Allocation of Satellite Capacity Through Packet Reservation," AFIPS Conference Proc. 42 (4-8 June 1973).

and some algorithm would be needed to control the direction of the one-way path. A simple technique would be for the speech controllers at either end of the conversation to alternate the transmission of SILENCE packets during periods when both parties were silent. During such a silent interval, either party could begin talking when it was his controller's turn to transmit the SILENCE packets. This technique for handing off the reservation would quantize the length of silent intervals to multiples of the round-trip time (0.27 sec). Since the speaker could begin talking at any arbitrary time, a delay of as much as twice the round-trip time could be added to the speech. Again, it might be more desirable to discard speech rather than introduce more delay.

In the case of a satellite conference, the delay in switching speakers could be avoided some of the time by distributing part of the function of the CHAIRMAN program to the individual conference controllers. In the situation in which the CHAIRMAN has decided who is to speak next before the current speaker has finished, it would be possible for the switch to occur immediately upon receipt of an "I AM FINISHED" message from the current speaker. Such distribution of control function should be possible in the satellite network because all sites would receive control messages simultaneously.

Using the handing off technique, we anticipate that speech delays in a satellite conference would be somewhat less than those observed in ARPANET conferences, but signaling delays would be greater. The effect of longer signaling delays should be most noticeable in the situations where a speaker has finished and no participant has yet indicated a desire to talk.

Since it appears that it will be some time before any actual packet satellite speech experiments can be undertaken, we propose to carry out some simulations which should help to evaluate the relative desirability of the approaches discussed above. Work is under way on programs to simulate the technique of handing off reservations. We are simulating this approach first because adequate data are not yet available on the delays to be expected in acquiring and relinquishing reservations under the various allocation schemes.

In the conferencing area, we also propose to simulate a conference with short speech delays and longer signaling delays such as we anticipate would occur in the packet satellite environment.

D. USER INTERFACE

1. Initial Version of Phrase Recognizer

An initial version of the speech recognition component of the voice-controlled network user interface has been implemented. The basic recognition strategy was outlined in the last SATS. Since that report, modifications have been made in segmentation, strong vowel detection, measurements, and syllable scoring; and a phrase scorer has been developed which allows recognition of a key phrase within a more or less arbitrary carrier sentence. The phrase recognizer was trained for recognition of connected digit sequences. A test was made where 11 different 4-digit sequences were considered admissible and a choice among these 11 sequences was required. For 97 utterances distributed among 9 male talkers (only 2 of whom had contributed to the training data), 93-percent correct recognition was achieved. Detailed descriptions of all parts of the current phrase recognizer and test results are given in NSC Note 91.

2. Probability Density Function Approximation for Syllable Scoring

Since the last SATS, a technique for deriving syllable scoring tables directly from simple and easily updatable statistics on measurements has been introduced. Prior to syllable or phrase scoring, the phrase recognizer processes an input utterance to yield a matrix, with the rows corresponding to detected strong syllables in the utterance, and the columns corresponding to a set of measurements made on each strong syllable. For each of these detected strong syllables, the syllable scorer must assign a score to every syllable in the pool of syllables contained in the set of expected phrases. Determination of this score requires an estimate of the conditional probability density function (pdf), $p_{R_j|H_i}(R_j|H_i)$ for each of the measurements R_j given each of the hypothesized syllables H_i. In our system, the required pdf estimates are now derived from 4 empirical statistics: the maximum, minimum, mean, and mean-square of each R_j for each H_i. Each pdf is approximated as a three-interval, piecewise-constant function which spans a slightly wider range than the observed (minimum, maximum) and has the empirical mean and mean-square values. A simple set of equations which transform the empirical statistics into the three-level pdf is given in NSC Note 91. Note that the 4 statistics used have chosen to be updatable. Given these 4 statistics and a knowledge of the number of samples on which they are based, the effect of additional samples on these statistics can be included without any other information about the past data. This updatability is essential for the interactive training procedure described below.

3. Phrase Scorer

The ultimate job of the phrase scorer is to decide which of the allowed phrases is most likely contained in the spoken utterance. In addition, the phrase scorer does not assume knowledge of when the key phrase occurs in the input utterance. Consequently, there are two main aspects of the phrase scorer: time alignment of the phrase and scoring of the phrase given its best alignment.

Each allowable phrase is represented as an ordered string of syllables. This syllable string is time-aligned with consecutive detected strong vowels in the input utterance; the phrase scorer tries all possible alignments. The score for any particular alignment is the average of the scores for the syllables in the phrase. The alignment having the largest score is chosen as the alignment for a phrase. The score for each phrase is taken to be this maximum score. The phrase having the largest score is the choice of the phrase scorer.

In conjunction with this basic strategy, the phrase score has some provision for inconsistencies in strong vowel detection. Occasionally, an important syllable in an utterance is called weak or not detected. Therefore, the syllable scorer ignores it and does not pass any information concerning it to the phrase scorer. This error occurs most often at the end of an utterance. The phrase scorer attempts to take into account this type of error by allowing an alignment where the last syllable in a phrase does not match an anchor point at the end of the utterance. Another type of error is an extra or unexpected strong vowel which causes misalignment of the phrase. This often occurs in the word "seven," where the second syllable was declared to be strong about 20 percent of the time. This type of problem is often predictable and can be alleviated by allowing additional representation for phrases. The strategy is to include the predicted extra strong vowel in the alignment, but not to score it. For example, there are two entries for the phrase "4759": one is "four-sev-five-nine" and the second is "four-sev-rdv-five-nine." When the syllable "rdv" is encountered, it is aligned with a strong vowel in the input utterance but does not contribute to the score. The score for "4759" is taken as the larger of the scores of its two representations. Similarly, there would be 4 representations for the phrase "1377," with 2 possible rdv's. We will refer below to these alternate syllable sequences for the same phrase as "incarnations."

4. Interactive Training Facility

A facility has been developed for on-line accumulation of all information necessary for scoring. This information is organized into a training file which contains an entry for each word on which the system has been trained. A "word" in this context is an utterance of one or more syllables (possibly more than one English word), and a phrase is taken to consist of a sequence of "words." Each word entry in the training file includes: (a) a syllabified spelling of the word, with stressed syllables marked; (b) information regarding the set of "incarnations," or different patterns of strong syllables, which have been observed in training; and (c) for each stressed syllable and each measurement, the maximum, minimum, mean, mean-square, and count of the samples observed so far.

To train the system on a new word entry, the utterance is spoken and is typed in a form which indicates syllabification and stress marks (as in a standard dictionary). Using a procedure similar to that described for the phrase scorer, the training system creates its best alignment of the typed-in stressed syllables with automatically detected strong syllables, and displays the results for the user. If the user is satisfied with this alignment, he types "OK," and the data for this spoken utterance are used to initialize a training file entry. Otherwise, he can interactively modify the alignment before typing "OK," or can choose to throw out this particular utterance. On subsequent repetitions of the utterance, the entry is updated by adding any new observed "incarnations," and incorporating the effect of the additional sample into each measurement statistic. After 2 or 3 repetitions of the same utterance, the syllabic alignment produced by the system is generally correct and training proceeds quite smoothly.

An important feature of these training files is that they can be easily combined yielding a composite training file with composite statistics. This feature results from the updatability of the measurement statistics, and can be exploited to study comparative performance of the phrase recognizer for various training conditions. For example, a talker can be tested against his own training data, or against composite data from several talkers which either includes him or does not include him. Also, the effect on performance of the number of repetitions included in training can be conveniently investigated.

To produce scoring tables usable by the phrase recognizer, the word sequences which make up the set of allowable phrases must be specified, and the measurement statistics in the training file must be transformed into pdf approximations using the technique discussed above. Training on words rather than on phrases allows the same word to be used in more than one phrase without retraining on the word. Also the set of allowable phrases can be modified with no additional training as long as no new words are used.

At this date, the interactive training facility has just become operational. A few informal experiments with single talker training have indicated that the method leads to essentially the same recognition results as the previous, much more laborious method which required extensive labeling of digitized speech. This interactive facility will be used heavily in training the phrase recognizer on the relatively large set of potential input utterances necessary for a real-voice-controlled network user interface.

5. Voice-Controlled ARPANET Mail Retrieval System

A system structure and a set of allowable voice inputs have been specified for an initial implementation of a voice-controlled ARPANET mail retrieval system. This system will provide the first demonstration of a voice-controlled user interface to a real network resource.

The initial implementation is designed to be as simple as possible while providing a realistic demonstration. Command text strings are delivered to TELNET as in the current mail retrieval system, but instead of being typed by the user, these command text strings are produced by a voice-controlled user interface in the local host (PDP-11/45) which translates speech to text. Only a limited subset of TENEX and MAILSYS commands is allowed, but the network TELNET, TENEX, and MAILSYS are used as they stand.

The voice-controlled user interface can be logically divided into three parts, as shown in Fig. I-1. The Task Controller sequences through a command structure stored in the local host which mimics the TELNET-to-TENEX-to-MAILSYS hierarchy. This Task Controller directs the Command Composer as to what commands are meaningful at each node in the command structure. The Command Composer is thereby enabled to help the user produce the desired command by a series of prompts and confirmations (initially in text form, but eventually as computer voice response). These aids are printed or displayed as text, and the user responds only by voice. The Command Composer also has the job of directing the phrase recognizer as to the next acceptable set of key phrases to be spoken by the user. Composed commands are transmitted to the Task Controller, which transmits these commands to TELNET, moves the user's position in the local control structure according to the received command, and directs the Command Composer as to the next acceptable command type. It is assumed that the local command structure and the command structure in the TENEX server stay in "sync." Maintaining "sync" is not anticipated to be a problem in this application.

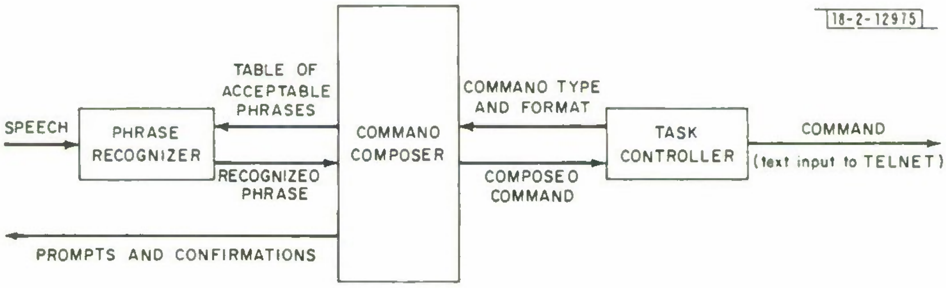

Fig. I-1. Voice-controlled user interface.

Eventually, it would be desirable for all information regarding the task and command structure to reside in the TENEX rather than in the local host. To accomplish this, a special TENEX executive and MAILSYS designed to interact with a voice user would have to be implemented. Conceptually, the Task Controller of Fig. I-1 would then reside in TENEX. This type of structure is more easily generalizable to other tasks but more difficult to implement. The simpler implementation described above should be sufficient for an initial demonstration of voice control of a network task.

Table I-1 gives an indication of the types of voice inputs which must be recognized in this application, and a possible correspondence between spoken "key phrases" and network commands. The most difficult job for the phrase recognizer will probably lie in recognizing parameters such as host name or message number rather than the commands themselves. Training of the phrase recognizer on a vocabulary appropriate for network mail retrieval is currently under way.

TABLE I-1

VOICE COMMANDS AND PARAMETERS

Command Type	Command	Key Phrases	Parameters
TELNET	CONNECT	make a connection	host name
	CLOSE	close the connection	none
	QUIT	leave voice control	none
TENEX LOGIN	LOGIN	log me in	log in name, account number, password
	LOGOUT	escape to TELNET	none
TENEX EXEC	MAILSYS	read my mail	none
	SYSTAT	system status	none
	DIR	list my directory	none
	LOGOUT	log out	none
MAIL RETRIEVAL	SURVEY	list the contents	none
	READ	type out	message number
	DELETE	delete	message number
	UNDELETE	restore	message number
	EXPUNGE	expunge	message number
	DESCRIBE	I need help	mail retrieval command name
	NEXT	next message	none
	QUIT	I am finished	none

E. PITCH EXTRACTION FOR DEGRADED SPEECH

A new frequency-domain pitch extraction algorithm has been developed which determines the pitch of the speech waveform from the spacing between the harmonics of the spectrum. The algorithm was designed for good performance when the input signal is noisy and/or distorted. Preliminary results indicate that the technique shows a substantial improvement over time-domain techniques when the input is telephone-quality speech or is corrupted with periodic noise. The algorithm, although more time consuming than the Gold-Rabiner method, operates well within real-time limits on the LDVT.

The algorithm is composed of three basic elements: the preprocessing to obtain a spectrum with adequate frequency resolution and bandwidth; the decision logic, to determine the most likely pitch choice based on the spacing between peaks in the spectrum; and the buzz-hiss decision. An important design consideration was that the algorithm be fast. The most time consuming part is the preprocessing, which requires 2.66 msec of time for each 10 msec spectral update, in the LDVT.

Thc spectral region decided upon runs from 210 to 1050 Hz. The ehoice of the lower limit was motivatcd by the fact that the information bclow 210 Hz is removed by a typical telephone filter, and the upper limit was chosen because the harmonics become increasingly ragged at higher frequencies. The resulting total range of 840 Hz is sufficient to assurc the presence of at lcast 2 harmonics of all but very high pitched voiced.

Thc preprocessing step makes use of some digital signal-processing techniques to obtain an adequate spectrum with a minimal requirement of computation timc and memory. In brief, the spectrum is first shifted such that 630 Hz is moved to the origin. By lowpass filtering the resulting complex signal to 420 Hz, and downsampling, one obtains a signal containing the spectral information from 630 – 420 to 630 + 420 Hz, as desired. In the proeess of shifting, filtering, and downsampling, a 9:1 reduction in data rate is aehieved. Thus, one ean obtain a speetrum with bctter than 7-Hz resolution, using only a 128-point FFT.

A new speetrum is obtained every 10 msec and, therefore, the piteh value is updatcd twiee per 20-msec frame. By thus oversampling thc pitch, onc ean make morc effective usc of mcdian smoothing techniques and, more importantly, ean gain a mueh better measure of the relative randomness of the piteh track as a major element in thc buzz-hiss decision. The pitch is ultimately transmitted only onee per frame.

The mcthod for cxtracting the piteh information from the peaks of the spcctrum is indieated schematically in Fig. 1-2. In general, during voieed segments, there is a set of equally spaced peaks of unequal size at the harmonics of the pitch. There are frequently extraneous peaks, particularly if the signal is degraded. The method uses an iterative teehnique whereby peaks are added one at a time, in order of descending size, to the set under eonsideration. After the addition of cach new peak, a new set of potential pitch estimates is added to a tablc of such cstimates, each cstimatc being defined as the distancc bctween adjacent peaks among the ones currently under consideration. After each iteration, a search is made for a sufficiently long list of "equal" (to within 14 Hz) estimates in the growing set of estimates. As soon as enough "equal" estimatcs are found, then the average valuc of these estimates is ehosen as the piteh. If the entire list of available peaks is cxhaustcd before an adequate score is obtained, then the average of thc best scoring set is seleeted as the pitch.

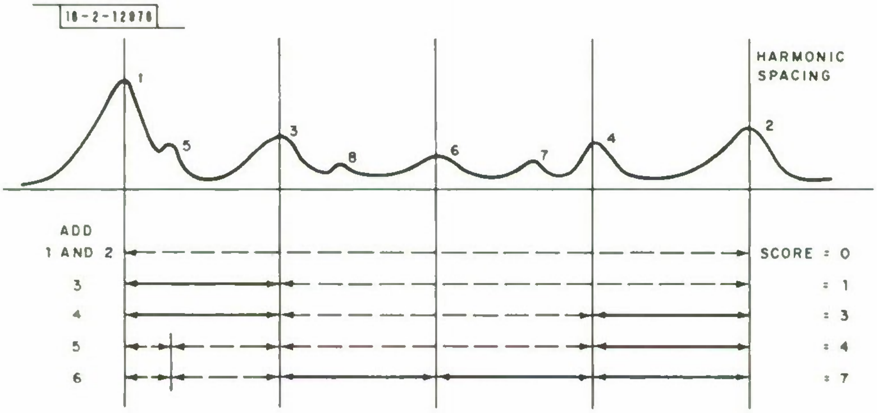

Fig. I-2. Diagram of seoring algorithm for extraeting piteh value from spaeing between harmonics. Correet spacings are shown as solid lines. Seore of number of eorreet ehoiees increases with eaeh iteration. Final seore is 7 after 5 interations.

Pitch values are obtained every 10 msec using the above technique. Then the resulting pitch contour is processed through a 3-point median smoother followed by a 5-point median smoother. It is at the step of median smoothing that the buzz-hiss decision is made. Specifically, if at least 2 out of 3 inputs of the 3-point smoother are "equal," the output is considered voiced. If, then, at least 3 out of the 5 inputs to the 5-point smoother are "equal," the final output is considered voiced. The only other parameter used for buzz-hiss is a very conservative silence threshold. It was specifically decided not to make use of any of the more usual buzz-hiss parameters, such as zero crossing density or energy ratios, becausethese parameters are quite sensitive to additive noise and filters such as a typical telephone channel filter.

We are encouraged by initial results of the performance of this harmonic pitch detector. We plan to make quantitative comparisons between its performance and that of our time-domain algorithm and other frequency-domain algorithms, using a variety of degraded input signals.

F. DIGITAL FILTER WITH VARIABLE CUTOFF

Lincoln and M.I.T. have completed the collaboration leading toward the design and construction of a variable cutoff lowpass digital filter. This filter (designed by A. P. Holt of M.I.T.) is a 128th-order finite-impulse response filter where the delay elements have been replaced by first-order allpass networks with an adjustable constant. Thus, by turning a single knob, the user can vary the filter cutoff frequency from as low as 200 Hz to as high as 10 kHz. Since the filter is of a high order (128), its characteristic very closely approximates that of the ideal rectangular lowpass. The filter is equiripple in both passband and stopband, with stopband sidelobes lower than −55 dB. The manual knob can be replaced by a computer input so that the cutoff frequency can be automatically varied, if so desired. This feature could be quite useful, for example, in a time-sharing environment where more than one user desires access to a real-time speech processor such as the LDVT.

G. CHARGE-COUPLED DEVICES

Two speech processor algorithms have emerged as possibly suitable vehicles for eventual CCD implementation. The channel vocoder is comprised to a great extent of fixed bandpass or lowpass filters. By fabricating several such filters on a chip using CCD techniques, it seems probable that an experimental CCD implementation can be realized which would have potential for economic mass production of such channel vocoders. Our present thinking would incorporate the channel filter banks into the CCD framework but allow the coding and pitch algorithms to be performed via low-speed microprocessor designs. The homomorphic vocoder algorithm, via the chirp-z transform, also allows for CCD implementation. In this case, pitch extraction might be more easily implementable since the cepstrum would be readily available.

Within the next several months, we expect to make decisions as to our implementation strategy; either channel vocoder, homomorphic vocoder, or both. Further analysis may show that the two algorithms may be merged into what might be called "spectrum-measuring" vocoders. At present, the thinking and discussion surrounding this important problem (which can lead to substantially lower cost speech terminals) is a 4-way joint effort of ARPA, M.I.T., Texas Instruments, and Lincoln Laboratory.

II. ACOUSTIC CONVOLVERS

A. INTRODUCTION

Acoustoelectric surface-acoustic-wave devices are being developed and their unique properties exploited as a means of implementing wide-bandwidth communications in the Packet-Radio System. These devices allow wideband spectral spreading, the use of continuously changing codes, and fast synchronization.

Lincoln Laboratory is constructing three sets of acoustoelectric modules to be delivered to Collins Radio during July 1976. Each set of modules contains two convolvers and the associated waveform generators, amplifiers, and switches to perform demodulation of differential-phase-shift-keyed (DPSK) data at a 100-kbit/sec rate. The DPSK modulation is imposed on a continuously changing PN coded carrier at either 100 Mchip/sec or 20 Mchip/sec. Final assembly and testing of the DPSK convolvers and the electronics packages is currently under way.

A fast synchronization scheme is being tested at Lincoln Laboratory. This scheme will minimize the preamble length, and will allow the use of continuously changing codes in the spreading sequence. This provides secure communications and protection against repeat jamming. The synchronization technique has been analyzed, and hardware for testing the technique is being assembled.

A coherent-integrator device is being developed for possible use as a means to overlay successive correlation spikes out of a convolver, thereby providing up to 30-dB additional correlation gain beyond that provided by a convolver. This would yield a maximum of 60-dB correlation gain. A system employing coherent integrators has been devised to provide variable correlation gain so that various S/N environments can be tolerated by the Packet-Radio System. Experimental results on the implementation of coherent integrators are expected during FY 7T.

B. CURRENT STATUS OF DPSK CONVOLVER SUBSYSTEM

Lincoln Laboratory has been developing acoustoelectric-convolver subsystems during FY 75-76 for the ARPA packet radio. A prototype convolver was developed during FY 75 for demodulating PSK spread-spectrum signals that were encoded with a continuously changing shift-register code with rates as large as 70 Mbits/sec. The convolver was delivered to Collins Radio together with a test set in August 1975. From the Fall of 1975 to the present, we have been developing a convolver subsystem for the demodulation of DPSK spread-spectrum signals. The initial design was for two parallel convolvers, each accommodating a signal with a time duration of 10 μsec and a bandwidth of 100 MHz. The convolver consisted of two parallel acoustic beams on a single lithium-niobate substrate, with a separate silicon strip for each convolver. This particular design was chosen because it provided 3 dB more dynamic range relative to an alternate method, in which the two silicon sections are placed in series on a single acoustic beam. By late Fall of 1975, it became clear that the parallel method would lead to unacceptable phase misalignments for DPSK demodulation because of phase errors which were caused by minuscule variations of several hundred angstroms in the gap between silicon and lithium niobate. The series convolver configuration was known to be insensitive to this kind of error. We, therefore, redesigned the convolver, and are currently assembling several series-beam convolvers. Initial test results show that 20-μsec-long convolvers can be reproduced with quite similar characteristics and we expect to deliver two subsystems to Collins this summer in time for their incorporation into the packet radio.

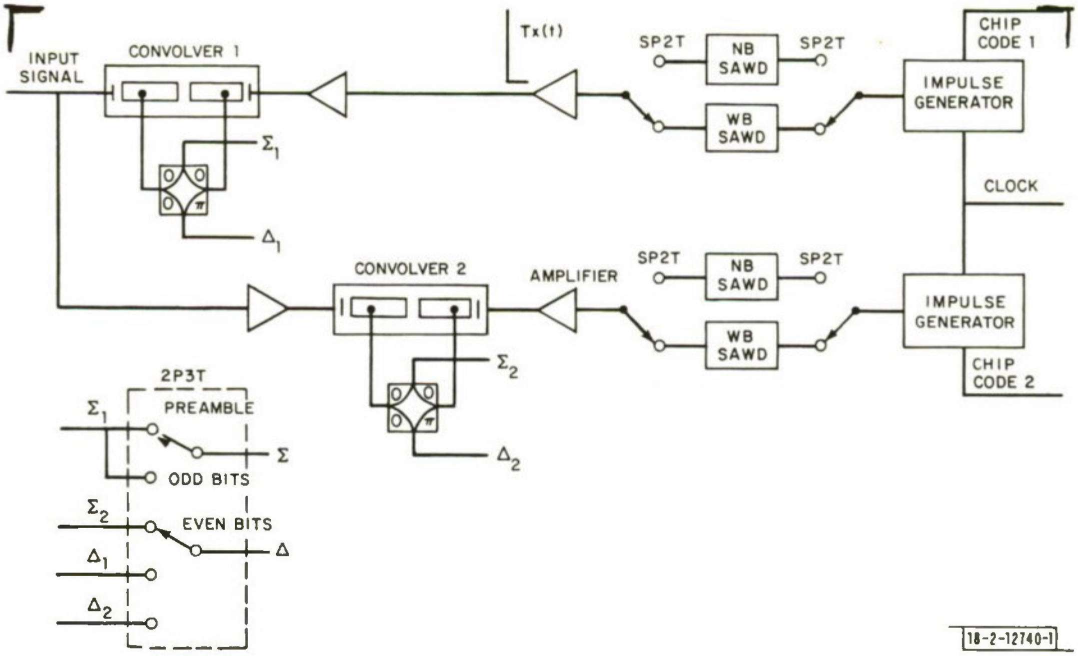

Fig. II-1. Convolver module schematic. The convolver module is used to interface the DPSK convolver with the input RF signal and digital chip codes. The module contains two SAW devices which are impulsed for either narrowband or wideband chip coding. An output switching network is used either to combine the convolvers during the preamble or to alternately select each of them during communications.

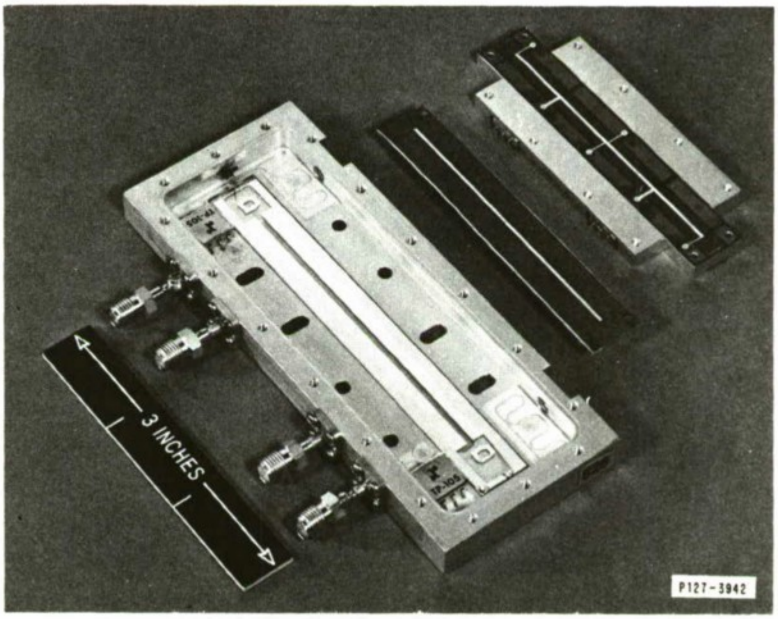

Fig. II-2. Three main convolver subassemblies. The base holds the $LiNbO_3$ delay line and matching networks. A Kapton sheet has the two silicon strips bonded to it. The Kapton circuit which provides the electrical interconnections to the silicon is attached to the pressure-plate assembly.

A block diagram of the convolver subsystem is shown in Fig. II-1. Phase-coded reference signals which are generated elsewhere in the packet radio trigger impulses which in turn create the reference waveforms on a 300-MHz carrier. Each 10-μsec-long bit is encoded with this continuously changing pseudorandom code. The incoming signal is convolved with the reference for the purpose of detecting phase transitions between adjacent bits.

There is a 12.8-MHz and a 92.4-MHz chip-rate option available. Each subsystem contains two DPSK convolvers, and each DPSK convolver has two input and two output terminals. One input is for the reference and the other for the signal. Each of the serially mounted silicon strips is connected to one of the output terminals, the outputs are connected to a hybrid, and the sum and difference ports of the hybrid are connected to a sensing network for the detection of phase transitions between adjacent bits. It takes 20 μsec to load a convolver with a reference. Consequently, each convolver senses one phase transition every 20 μsec. The reference signals into the two convolvers are interlaced in order to detect phase transitions every 10 μsec. The input terminal supplies a +10-dBm signal for each convolver. The output of the phase transition detection network is available at the module output terminal at a nominal power level of −50 dBm.

The DPSK convolver had to be designed to meet the packaging requirements of the packet radio. Two convolvers together with their output circuitry had to fit into a single packet-radio module measuring $4\frac{7}{8} \times 6\frac{1}{4} \times \frac{1}{2}$ in. The drive circuitry, the waveform generation circuits, etc., fit into a second unit of approximately the same size.

The initial prototype convolver worked over a limited temperature range, partly as a result of the large thermal expansion coefficient of RTV gel. The gel is necessary to establish and maintain a uniform 3000-Å gap between silicon and lithium niobate. The DPSK convolver is designed to compensate for the thermal expansion coefficient of RTV gel with an array of beryllium copper springs.

The sequence of Figs. II-2, -3, and -4 shows the design details of the convolver. Figure II-2 shows the lithium-niobate substrate with surface-wave transducers on each end of the substrate. Each transducer is surrounded with a grounded shield. Acoustic beams are propagated along the center of the substrate in opposite directions between the aluminum-gold ground planes on the lithium-niobate substrate. The open region between the ground planes contains a pseudorandomly distributed array of spacer posts. Two 1-in.-long strips of silicon are indium bonded to a 3-mil-thick Kapton sheet. There are electrical feedthroughs behind the silicon strips and ohmic contact is made to a Kapton microstrip line, which provides 30-ohm terminations and a central connection to each silicon strip to carry the output signals to impedance transformers. This microstrip line is attached to a mounting block, and the Kapton sheet with the silicon is placed over the microstrip. The mounting block is placed over the lithium-niobate delay line, and bolted in place as shown in Fig. II-3. The back side of the block has an open slot in which the back of the microstrip line can be seen. The RTV-gel strip is inserted in the slot over the microstrip, a thin strip of beryllium-copper sheet is placed over the gel, the spacer block for holding the springs is placed over the strip, and the springs are inserted inside the holes. A cover plate is then bolted over the springs. There are quarter-wave microstrip sections at the two input terminals. There are also impedance transformers at the input and output terminals to provide wideband impedance match to 50-ohm-coaxial cable. The completed package is sealed with covers as shown in Fig. II-4. As a final assembly step, two special ports in the device are opened, dry nitrogen is flowed through the device, and the ports are closed and sealed to protect the convolver from water vapor and dust.

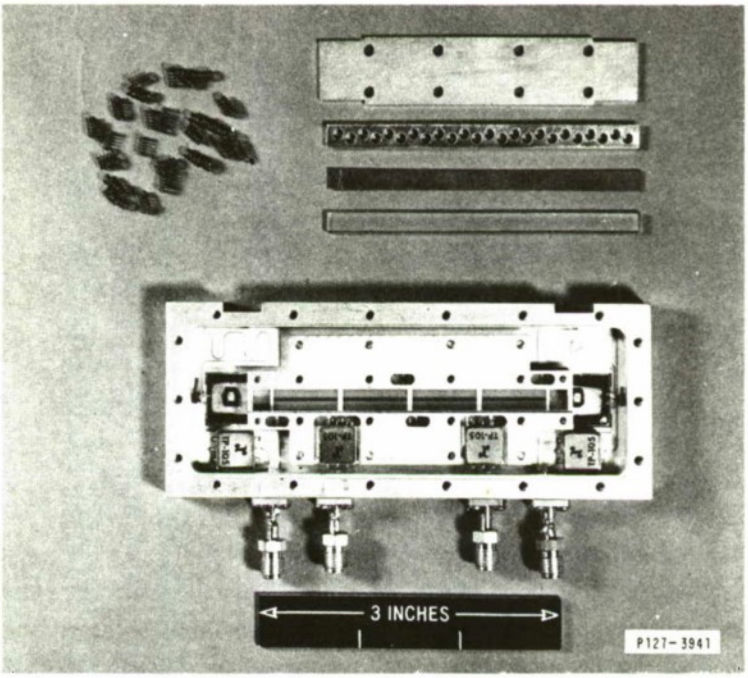

Fig. II-3. Convolver with silicon and pressure plate in place. The RTV and spring assembly are also shown.

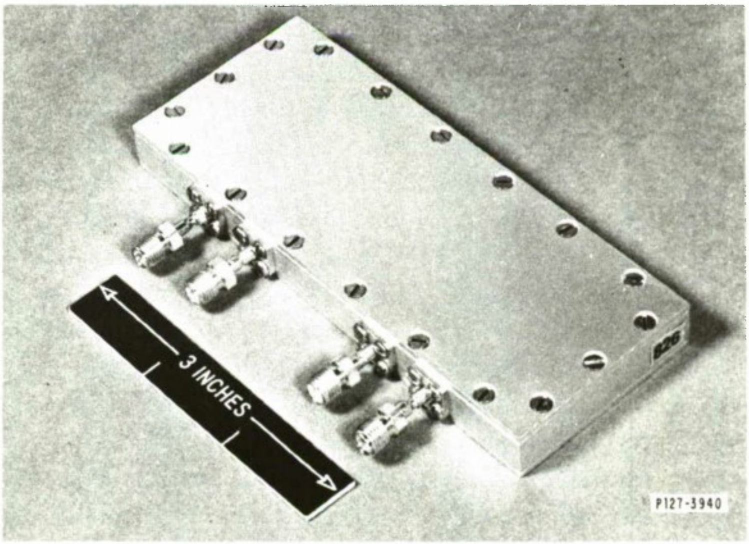

Fig. II-4. Assembled convolver.

The most critical assembly step is the placement of the silicon on top of the lithium-niobate spacer posts. A special jig developed for this purpose is shown in Fig. II-5. Here the technician aligns the carefully cleaned lithium niobate with the silicon strips. Once they are in position, the lithium niobate is guided down against the silicon strips by the jig. The gap between silicon and lithium niobate is visible through the open slots in the metallic base which contains the lithium niobate.

The performance specifications of the DPSK convolver are listed in Table II-1. Lincoln Laboratory is currently assembling six units meeting the specifications also shown in Table II-1. The electronic package is shown in Fig. II-6.

C. FAST LATCH-UP CIRCUIT

Lincoln Laboratory has been studying the problem of latching the DPSK-convolver subsystem to an incoming packet with a minimum preamble sequence with no repeated codes. A circuit diagram for this purpose is shown in Fig. II-7. Suppose a preamble is expected having bit sequences C_1, C_2, C_3, etc., in which each bit is encoded with a continuously changing pseudorandom code comprising a bandwidth of 100 MHz. Reverse-code sequences $C_2^-C_1^-C_2^-C_1^-C_2^-C_1^-$ and $C_3^-C_2^-C_3^-C_2^-C_3^-C_2^-$ are propagated through convolvers A and B, respectively. The incoming signal is fed into both convolvers. When preamble signal C_1C_2 lines up with the $C_2^-C_1^-$ reference, then a correlation impulse is obtained in convolver A. Ten microseconds later, a sampling gate is opened for the expected correlation impulse at convolver B. If the impulse is sensed, then code sequence $C_6^-C_5^-$ is entered into convolver A. If the correlation impulse is not obtained, then code sequences $C_3^-C_2^-$ and $C_4^-C_3^-$ continue to be circulated through convolvers B and A, respectively, until a correlation impulse is detected in convolver B. Thus, the system can recover from false alarms in convolver A. The use of a sampling gate to provide a second vote improves the probability of detection, and decreases the probability of false alarm. Refinements in the positioning of the reference-code sequence in the convolver subsystem can be made quite accurately after the first correlation impulse by delaying subsequent reference-code sequences appropriately. The probability of one or more false alarms in a 1-msec synchronization interval is calculated below.

In convolver A, the probability that noise will exceed the threshold during the synchronization interval is given by the Poisson distribution

$$P(K, \lambda_1) = \frac{e^{-\lambda_1 N} (\lambda N)^K}{K!}$$

where

K = number of times noise exceeds threshold,

$\lambda_1 = P_{fa}$ in a correlation cell in convolver A,

N = total number of correlation cells in the synchronization interval.

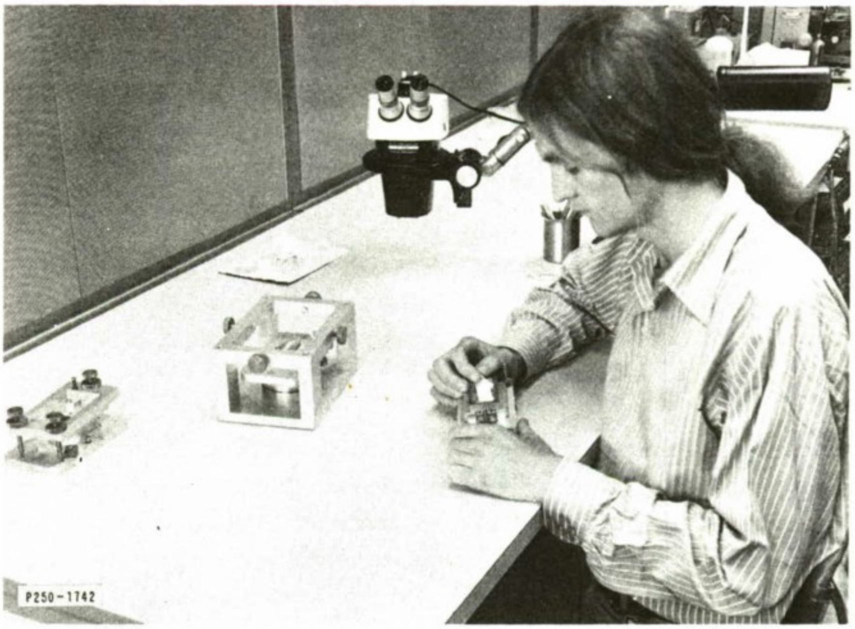

Fig. II-5. Technician preparing to use convolver assembly jig.

TABLE II-1
CONVOLVER SPECIFICATIONS

	Prototype	DPSK
Bandwidth	100 MHz (1-dB bandwidth)	100 MHz (1-dB bandwidth)
Convolution Interval	10 μsec	20 μsec
Time-Bandwidth Product	1000	2000
Convolution Uniformity	± 1/2 dB	± 1/2 dB
Dynamic Range	50 dB (+14 dBm inputs maximum)	40 dB (+15 dBm reference)
Spurious Signals	>40 dB down from desired signal	40 dB down from desired signal
VSWR	<3:1 (all ports)	<3:1 (all ports)
Temperature Range	−10 to +30°C	−25 to +50°C

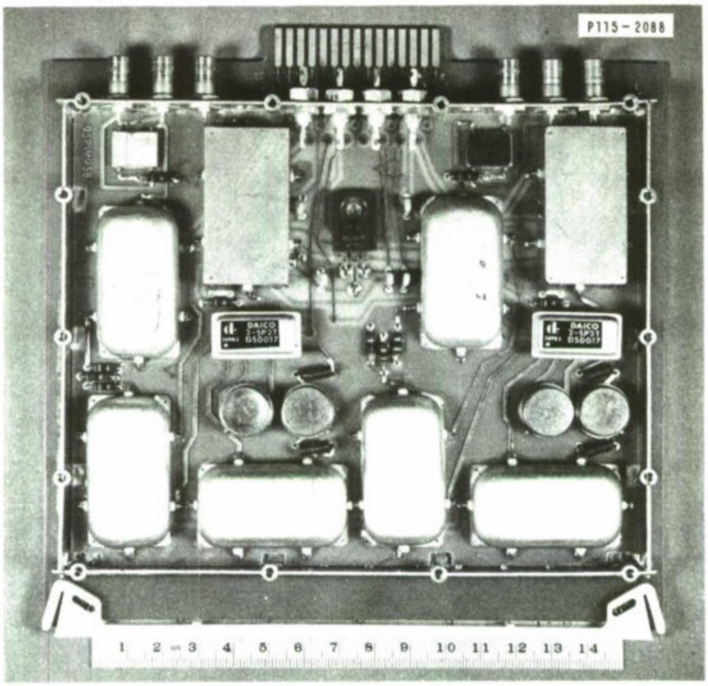

Fig. II-6. Portion of the electronics to be used with the convolvers in the Packet-Radio System.

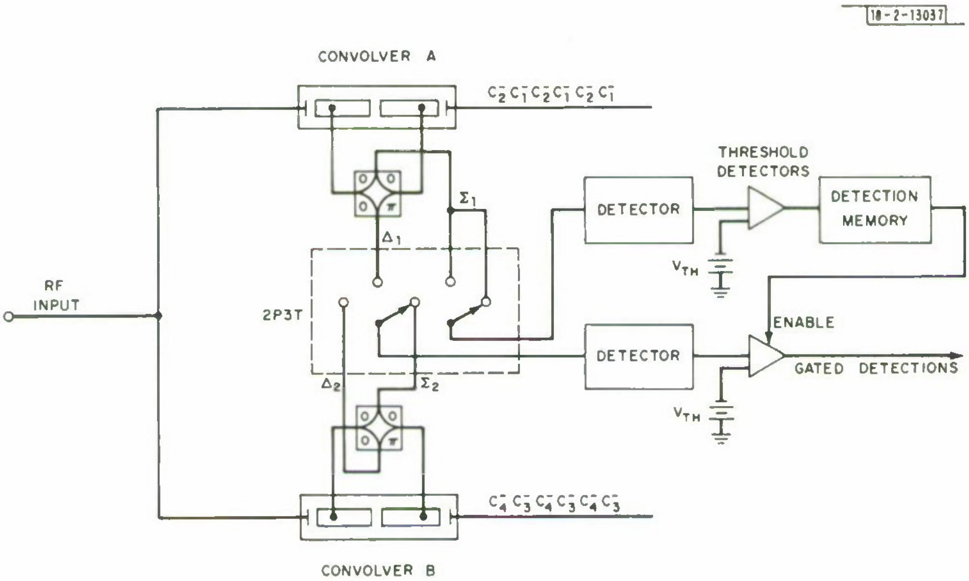

Fig. II-7. Block diagram of a synchronization system for packet-radio applications. Detections sensed in convolver A are delayed by two bit periods in the detection memory. When a detection in convolver B coincides with the detection-memory output, a packet is deemed present and the communications cycle can begin.

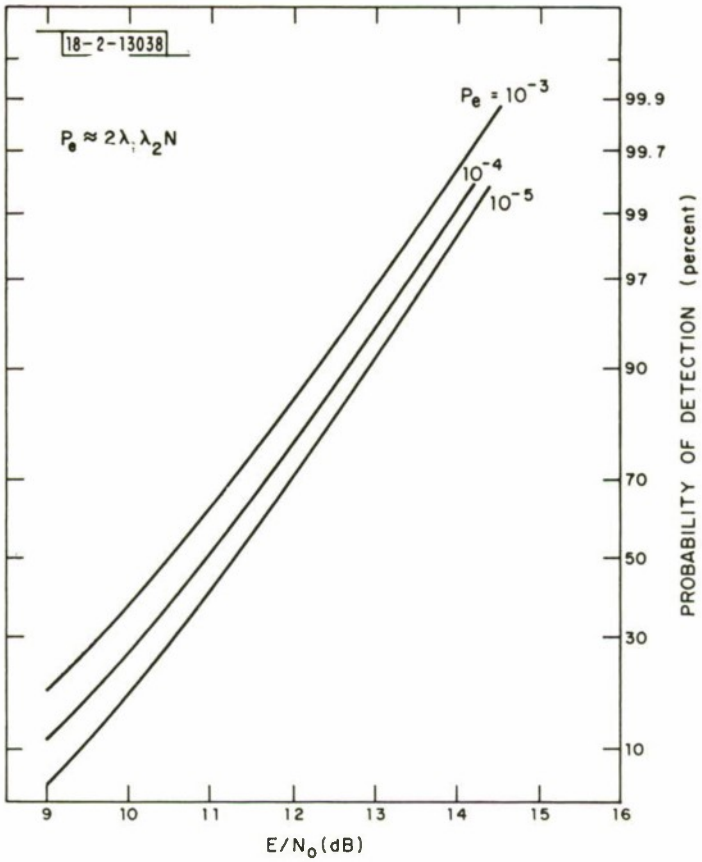

Fig. II-8. Probability of detection vs signal-to-noise ratio for successive detections in convolvers A and B.

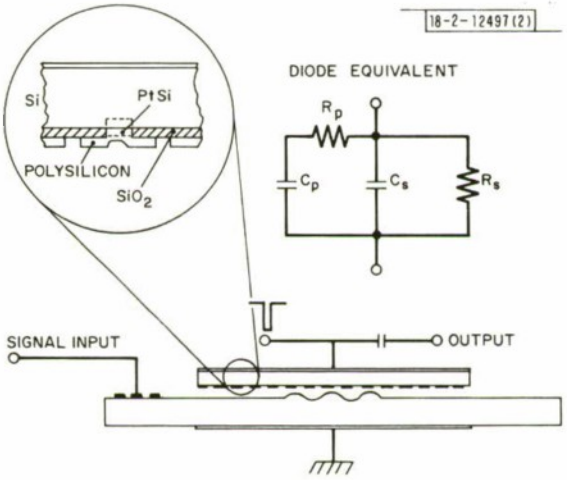

Fig. II-9. Coherent integrator schematic and equivalent circuit of diodes.

In convolver B, the probability that noise exceeds the threshold is a conditional probability which depends on the number of times noise exceeds the threshold in convolver A. This probability of error is:

$$P_{error} = P(\text{error in convolver B/1 error in convolver A})$$
$$+ P(\text{error in convolver B/2 errors in convolver B})\ldots$$

$$P_{error} = [1-(1-\lambda_2)^2]\ P(1,\lambda_1) + [1-(1-\lambda_2)^4]\ P(2,\lambda_1) + \ldots$$

$$P_{error} = \sum_{i=1}^{N} [1-(1-\lambda_2)^{2i}]\ P(i,\lambda_1) \quad .$$

This series can be closely approximated by:

$$P_{error} \approx 2\lambda_1\lambda_2 N \qquad \lambda_1,\ \lambda_2 << 1 \quad .$$

Figure II-8 is a plot of the probability of detection vs output signal-to-noise ratio with probability of error as a parameter. If an operating point for communication is selected so that γ = 14.2 dB, then the signal-to-noise ratio for synchronization can vary from 13 to 14.2 dB depending on the relative code alignment. For $P_{error} = 10^{-3}$ (one synchronization interval in 1000 has a false packet decision), the probability of detection is 0.97 for the worst case of code misalignment (i.e., γ = 13 dB). For the average case of code misalignment (γ = 13.6 dB), the probability of detection is 0.99. During communication, the probability that a bit will be decoded with error is 10^{-6}. The probability that one or more bits in a 1000-bit packet will be decoded with error is

$$P_e\ (\text{packet}) = 1-(1-10^{-6})^{1000} = 1 \times 10^{-3} \quad .$$

On the average, only one packet in 1000 will be decoded with error. This circuit in Fig. II-7 is currently being assembled, and initial results are expected during FY 7T.

D. COHERENT INTEGRATOR

M.I.T. Lincoln Laboratory has been exploring novel means to provide correlation gains in excess of 10^5 with continuously changing spread-spectrum codes. During FY 76, we have explored the feasibility of a coherent integrator.

A schematic diagram of the coherent integrator is shown in Fig. II-9. The coherent integrator consists of a substrate of lithium niobate with an adjacent strip of silicon. The silicon is covered with a two-dimensional array of Schottky diodes, whose contacts are overlaid with squares of polysilicon, as can be seen in the figure. The equivalent circuits for the diodes is shown as well. The piezoelectric field associated with the surface acoustic waves causes image charges to flow into the Schottky diode capacitance C_s^+ during the instant of forward bias. The diodes are then biased in the reverse direction, which reduces the diode capacitance C_s^- by a factor of 10. The decrease in capacitance causes an increase of potential across C_s, and the potential causes a current to diffuse into the polysilicon capacitance C_p. The polysilicon has a time constant of R_pC_p, which is of the order of a microsecond. A succession of signals occurring as frequently as every 10 µsec could be sampled in this fashion, with the result that the image charges are accumulated in the polysilicon capacitor C_p until the potential across C_p equals the potential across C_s.

The number of charge samples that could be accumulated in a coherent integrator is proportional to C_p/C_s^-. Ratios as large as 50 to 100 appear technically feasible by this method. This calculation is based on the assumption that the sampling impulse which forward-biases the Schottky diodes is on for a sufficiently long time to fully charge the diode capacitance C_s^+. It may become possible to reduce the time duration of the sampling impulse, so that only a portion of the piezoelectric image charge appears in C_s^+. Typically, a full image charge is obtained within 3 nsec. If we could realize a switching circuit with a variable time gate ranging from 0.3 to 3 nsec, then it should be possible to vary the time duration of the storage impulse and thereby increase the number of possible overlays to as many as 1000.

Solid-state versions of a fast switching circuit appear feasible, providing no more than 50 V are required. The forward-biasing potential required by the diodes is less than a volt. However, the potential drop across the lithium-niobate substrate, which is in series with the Schottky diodes, is 10 V per mil of lithium-niobate thickness. Consequently, a thin 5-mil-thick substrate of lithium niobate is needed to reduce this potential to a reasonable level. We have been exploring a technique whereby a 5-mil-thick wafer of $LiNbO_3$ is glued to a thick substrate of lithium niobate. The wafer-substrate interface is coated with a transparent conductor to provide a ground electrode.

Schottky diodes have been fabricated on 12-μm centers that appear to have a storage time in excess of 100 msec. Thus, it seems feasible that a coherent integrator should be possible for overlaying as many as 1000 10-μsec-long signals. Furthermore, if these signals have a repetitive code 10-μsec long, then it is possible to read out this overlaid code with a reference signal. This would provide an additional correlation gain of 30 dB and a net maximum signal-processing gain of 60 dB.

Reading out the integrated signal by correlating with a reference signal would be highly constraining to the user because the same code sequence would have to be repeated many times. This could lead to multipath problems, difficulties with repeater jammers, and reduced securit). If it could be assumed that the timing of the incoming signal is known within about 1 μsec, then a convolver with a continuously changing reference code could decode the signal, and incidentally provide a correlation gain of 30 dB. The convolver output would then be overlaid in a coherent integrator to provide as much as 30 dB of additional signal-processing gain. The overlaid correlation impulses could then be read out. For maximum dynamic range, the readout could be implanted with a coded read-out waveform. A subsystem which performs these functions would require a generator for the reverse-order reference code, an acoustoelectric convolver, a coherent integrator, a precision pulser, a coded waveform generator and its matched filter, and the usual amplifiers, local oscillators, and mixers.

At present, the piece parts of a narrowband version of the coherent integrator are available. During the remainder of FY 76 and FY 7T, we expect to assemble the coherent integrator and perform some initial tests with it. Some results should be available early in FY 7T.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS
BEFORE COMPLETING FORM

1. REPORT NUMBER
ESD-TR-76-186

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE *(and Subtitle)*
Information Processing Techniques Programs:
Packet Speech/Acoustic Convolvers

5. TYPE OF REPORT & PERIOO COVERED
Semiannual Technical Summary
1 January – 30 June 1976

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR*(s)*
Bernard Gold and Ernest Stern

8. CONTRACT DR GRANT NUMBER*(s)*
F19628-76-C-0002

9. PERFORMING ORGANIZATION NAME ANO AOORESS
Lincoln Laboratory, M.I.T.
P.O. Box 73
Lexington, MA 02173

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
ARPA Orders 2006 and 2929
Program Element Nos. 62706E and 62708E
Project Nos. 6P10 and 6T10

11. CDNTRDLLING OFFICE NAME AND ADDRESS
Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

12. REPORT DATE
30 June 1976

13. NUMBER OF PAGES
28

14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*
Electronic Systems Division
Hanscom AFB
Bedford, MA 01731

1S. SECURITY CLASS. *(of this report)*
Unclassified

1Sa. DECLASSIFICATION OOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

packet speech
network speech
syllable scoring
convolvers
coherent integrator
probing structures
differential rate
C^2 links
adaptive routing
time-varying communications
ARPANET

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This report describes work performed under three programs: Packet Speech, Acoustic Convolvers, and Airborne Command and Control sponsored by the Information Processing Techniques Office of the Defense Advanced Research Projects Agency during the semiannual period 1 January through 30 June 1976. The first two programs are reported in Vol. I and the third in Vol. II.